(12)

A RAND NOTE

AD A113583

TATR: AN EXPERT AID FOR TACTICAL AIR TARGETING

Monti Callero, Lewis Jamison,
D. A. Waterman

January 1982

N-1796-ARPA

Prepared For

The Defense Advanced Research Projects Agency

DTIC
APR 1 9 1982
H

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

DTIC FILE COPY

Rand
SANTA MONICA, CA. 90406

The research described in this report was sponsored by the Defense Advanced Research Projects Agency under ARPA Order No. 3460/3473, Contract No. MDA903-78-C-0029, Information Processing Techniques.

The Rand Publications Series: The Report is the principal publication documenting and transmitting Rand's major research findings and final research results. The Rand Note reports other outputs of sponsored research for general distribution. Publications of The Rand Corporation do not necessarily reflect the opinions or policies of the sponsors of Rand research.

Published by The Rand Corporation

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER N-1796-ARPA	2. GOVT ACCESSION NO. AD-A113	3. RECIPIENT'S CATALOG NUMBER 583
4. TITLE (and Subtitle) TATR: An Expert Aid for Tactical Air Targeting		5. TYPE OF REPORT & PERIOD COVERED Interim
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Monti Callero, Lewis Jamison, D. A. Waterman		8. CONTRACT OR GRANT NUMBER(s) MDA903-78-C-0029
9. PERFORMING ORGANIZATION NAME AND ADDRESS The Rand Corporation 1700 Main Street Santa Monica, CA. 90406		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Advanced Research Projects Agency Department of Defense Arlington, VA 22209		12. REPORT DATE January 1982
		13. NUMBER OF PAGES 45
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release: Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

No Restrictions

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Artificial Intelligence
Tactical Bombing
Targets
Military Air Facilities
Heuristic Methods

DTIC ELECTE APR 19 1982 S D H

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

See Reverse Side

DD FORM 1 JAN 73 1473

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Describes an initial version of TATR (tactical air target recommender), a prototype "expert system" being developed at Rand to assist tactical air targeteers in selecting and prioritizing targets. TATR applies a knowledge engineering problem-solving approach in which human domain knowledge is essential, and judgment plays a larger role than mathematical algorithms and stochastic formalisms. Under interactive user direction, TATR preferentially orders enemy airfields, determines targets on those airfields to attack, and identifies the most effective weapon systems against those targets. TATR is programmed in the ROSIE language, which has an English-like syntax that facilitates nonprogrammer comprehension and program verification. The program applies predetermined planning heuristics to generate an airfield attack plan. It then replans, incorporating user modifications, and projects a series of plans over several days. TATR interactively maintains databases by requesting and processing updates from the user and provides detailed information about plans, friendly force capability, and enemy force posture and status.

A RAND NOTE

TATR: AN EXPERT AID FOR TACTICAL AIR TARGETING

Monti Callero, Lewis Jamison,
D. A. Waterman

January 1982

N-1796-ARPA

Prepared For The Defense Advanceu Research Projects Agency

Rand
SANTA MONICA, CA. 90406

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

PREFACE

Under the support of the Information Processing Techniques Office of the Defense Advanced Research Projects Agency, Rand has been investigating the possibility of applying new technology in the field of artificial intelligence to the problem of Air Force tactical planning. This research has focused on the possibility of using the tools and techniques of knowledge engineering to construct an intelligent assistant "expert system" for tactical air targeting. This Note describes the initial version of such an expert system. The system, called the tactical air target recommender (TATR), was developed by Rand, with input from professional Air Force targeting personnel. Although only the first step in an evolutionary development, this version of TATR should be of interest to tactical planners and to practitioners and researchers developing either expert systems or automated aids for tactical planning.

DTIC
COPY
INSPECTED
2

Accession For
NTIS GRA&I ☑
DTIC TAB ☐
Unannounced ☐
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A

SUMMARY

Rand is developing a prototype "expert system" to help tactical air targeteers select and prioritize airfield targets. The system, called the tactical air target recommender (TATR), applies a knowledge engineering problem-solving approach in which human domain knowledge is essential, and judgment, experience, and intuition play a larger role than mathematical algorithms and stochastic formalisms. Based on collective inputs of experienced Air Force tactical air targeteers, TATR performs the following tasks under the interactive direction of a user: preferential ordering of enemy airfields; determination of the targets on those airfields to attack; and identification of the weapon systems that can be most effective against those targets.

A key feature of TATR is that it is programmed in the ROSIE programming language. ROSIE was specifically designed by Rand to support knowledge-based programming tasks. It readily accommodates heuristic logic and has an English-like syntax that facilitates non-programmer comprehension and verification of the program. The readability aspect also enables the user to readily determine program modifications as the knowledge base evolves. Hence, TATR can provide a vehicle for the development and evolution of targeting concepts and approaches.

TATR is an interactive program that performs its functions and produces outputs only at the direction of a user. Its primary function is to provide the user with a plan for attacking enemy airfields and to project the effects of implementing the plan. The plan results from a

joint user/program interchange. The program applies predetermined planning heuristics to generate an initial plan that can be modified by user guidance or specific instructions. The program then replans to incorporate the user's directions. By projecting the results of a series of plans over a number of days, TATR can aid the user in deciding on the best plan or sequence of plans to implement.

In addition to the basic planning function, TATR also interactively maintains its databases by requesting and processsing updates from the user and, in response to user requests, provides detailed information about plans, friendly force capability, and enemy force posture and status.

Our objective is to develop TATR as a prototype expert system with sufficient expert knowledge and functional capability to be transferable to the Air Force for operational experimentation and development as a targeteer's aid. The initial version of TATR described in this Note is the first step in an evolutionary process which typifies the development of knowledge engineering systems. At each iteration, TATR will provide the vehicle for stimulating new perceptions and articulations of the targeting task, which will then become the basis for the next iteration. This process is now in progress.

CONTENTS

FIGURES

TABLE

I. INTRODUCTION

In wartime, the tactical air planning process determines the intended operational use of tactical air resources in a future operational time period (usually the next day) and prepares the necessary orders and instructions for operational units, such as fighter wings, to execute the planned missions. The selection of enemy targets to be attacked is a core task in the planning process. Like most of the process, target selection depends predominantly on human judgment to integrate information about friendly and enemy force posture, capability, operations, and objectives to determine effective, efficient courses of action. Human decisionmaking is inherently unstructured, and its predominance in target selection has previously inhibited the development of automation tools to support this process. Because we believe that automated aids specifically designed to reflect the observable human decision process can contribute to better judgments, we are developing a prototype "expert system" to help tactical air targeteers select and prioritize targets. The program, called the tactical air target recommender (TATR), applies a knowledge engineering problem-solving approach in which human domain knowledge is essential, and judgment, experience, and intuition play a larger role than mathematical algorithms and stochastic formalisms.

Knowledge engineering requires many iterations of system implementation. The knowledge that human experts possess is often difficult to articulate because it may be incomplete, indefinitive, or inconsistent. Translating such knowledge into computer programs

produces precise and rigorous interpretations which lead to deeper understanding and new perceptions about the problem domain. These in turn stimulate changes in the knowledge base that translate into new, precise, and rigorous interpretations in the program. Hence, system development requires an evolutionary approach.

The version of TATR reported in this Note is the first step in such an iterative process.[1] Based on the initial collective inputs of experienced Air Force tactical air targeteers, the program performs the following tasks under the interactive direction of a user:

- Preferential ordering of enemy airfields.
- Determination of the targets on those airfields to attack.
- Identification of the weapon systems that can be most effective against those targets.

Further, it updates the status of database elements either from user inputs or from projections of the effects of friendly air operations. Follow-on iterations will incorporate better decisionmaking heuristics more closely emulating human targeteers, expanded interactive techniques to improve overall usability, and greater flexibility to adapt to new knowledge or unplanned-for conditions.

A key feature of TATR is that it is programmed in the ROSIE programming language. ROSIE (Rule Oriented System for Implementing Expertise) was specifically designed by Rand to support knowledge-based

---

[1] Previous prototype development efforts which contributed to the development of TATR are reported in Callero, Gorlin, Hayes-Roth, and Jamison (1981). That Note also describes the tactical targeting process and the knowledge engineering problem-solving approach.

programming tasks. It readily accommodates heuristic logic and has an English-like syntax that facilitates non-programmer comprehension and verification of the program. The readability aspect also enables the user to readily determine program modifications as the knowledge base evolves. Hence, TATR can provide a vehicle for the development and evolution of targeting concepts and approaches.

Our ultimate objective is to develop TATR as a prototype expert system with sufficient expert knowledge and functional capability to be transferable to the Air Force for operational experimentation and development as a targeteer's aid.

Section II describes the functions TATR performs, the outputs it produces, and the user actions and options it provides for. Section III describes the structure and logic of the main decisionmaking routines, and Section IV presents concluding comments. The Appendix contains an annotated trace of a user's scope during an example run.

II. TATR FUNCTIONS, OUTPUT, AND INTERFACE

OVERVIEW

TATR is an interactive program that performs its functions and produces outputs only at the direction of a user. Its primary function is to provide the user with a plan for attacking enemy airfields and to project the effects of implementing the plan. The plan results from a joint user/program interchange. The program applies predetermined planning heuristics to generate an initial plan that can be modified by user guidance or specific instructions. The program then replans to incorporate the user's directions, and so forth. By projecting the results of a series of plans over a number of days, TATR can aid the user in deciding on the best plan or sequence of plans to implement.

In addition to the basic planning function, TATR also interactively maintains its databases by requesting and processing updates from the user, and, in response to user requests, it provides detailed information about a plan, friendly force capability, and enemy force posture and status.

To facilitate understanding of TATR functions, outputs, and interfaces, and to introduce key definitions and terminology, we will sketch the plan-generation function and briefly discuss database information. A more extensive discussion of the program logic, heuristics, and calculations follows in Section III.

Plan-Generation Function

The program uses a database of information describing the airfields and the types of friendly forces available for the attack. When a user calls upon TATR to execute the plan-generation function, the program first generates an airfield target list (ATL) containing the most threatening enemy airfields. The threat of each airfield is indicated by a threat index (TI) based on the number and potential capability of each type aircraft located at the airfield. The ATL is next augmented to reflect perceived threat characteristics that are not reflected by the TI alone. Identification of airfields with such characteristics may either be resident in the database or input by the user at run time. They are referred to as "key unit" (KU) airfields.

Once the ATL is determined, the program "weaponeers"[1] each target element (e.g., aircraft, maintenance facilities, runways) at the airfields on the ATL and calculates the potential relative worth of attacking each of them. This number, the achieve ratio (AR), is the ratio of the expected reduction in the TI resulting from an attack on the target element to the number of aircraft needed to achieve an acceptable level of damage against that element. The program then orders the airfields on the ATL by a heuristic combination of the highest AR calculated at an airfield and the key unit status of the airfield. The results are displayed to the user.

[1] The weaponeering process can either (1) identify weapon systems that are effective against a target and determine the number of those weapon systems necessary to achieve a specified damage expectancy on the target, or (2) calculate damage expectancy against a target, given a specified type and number of weapon systems. Both approaches are used in TATR; the present discussion illustrates the former.

The ordered list of airfields, the target elements on each airfield determined best to attack, and the number and types of aircraft to assign to the attacks comprise the product of the plan-generation function. This plan information and all intermediate results (e.g., results of attacking other than the preferred target elements) are provided to the user in as much or as little detail as he or she specifies.

The Database

The TATR program requires a database consisting of general information about enemy airfields and specific information about the composition of particular enemy airfields. The general information includes lists of the types of targets that might be found on an enemy airfield, the types of enemy aircraft that might be present, the types of friendly aircraft and weapons available to attack the enemy airfields, and parameters of weapon system capability and effectiveness of friendly aircraft. The specific information includes a detailed description of each enemy airfield, usually containing 30 or more assertions describing the airfield. An example of the airfield information is shown in Fig 1.

The initial TATR database is generated in advance and updated dynamically as the program is being executed. This dynamic updating capability is an absolute necessity, since the status of the enemy airfields changes during the battle. Changes result both from enemy operations that diminish resources (consumption and attrition) and increase resources (replenishment and reconstitution) and from actions by friendly forces.

```
Let the name of A8 be "Falkenberg"
and the BE-number of A8 be 9030
and the location of A8 be <5132,1313>
and the ceiling at A8 be 14000 ft
and the visibility at A8 be 12 nm
and the number of landing_surfaces at A8 be 1
and the number of cuts to disable_landing_surfaces at A8 be 2
and the number of badgers at A8 be 54
and the number of fishbeds at A8 be 24
and the number of revetments at A8 be 10
and the number of shelters at A8 be 30
and the number of maintenance_hard at A8 be 3
and the number of maintenance_soft at A8 be 6
and the number of above_ground_pol_storage_sites at A8 be 6
and the number of below_ground_pol_storage_sites at A8 be 3
and the percent of pol in above_ground_pol_storage_sites at A8 be 65 %
and the percent of pol in below_ground_pol_storage_sites at A8 be 35 %
and the number of munition_sites at A8 be 5
and the percentage of munitions in largest_munitions_site at A8 be 25 %
and the sam_sites near A8 be <S1>
and the number of sam_sites at A8 be 1
and the distance to A8 be 80 nm
and the distance_to_target factor of A8 be 1.2
and create an air defense at A8.
```

Fig. 1--Example Data Entry for an Enemy Airfield

Changes in the enemy's status are recognized and processed in two different ways. The first and most reliable way is for friendly intelligence systems to observe and report changes. The information is extracted from the intelligence reports and entered by the user prior to or during interactive plan development. The second way is for TATR to recognize that friendly actions were previously planned to have taken place against an enemy airfield; hence, changes must have occurred but have not yet been reported. In this case, the user can enter the actual (if known) number and type of weapon systems and target elements attacked, and the program calculates the effects and reflects them as status estimates in the database. If no actual information about a

previously planned attack is available, the effects of the planned attack (as if it went as intended) are projected and reflected as estimates in the database. Reported information takes precedence over projected effects.

OUTPUT AND INTERFACE

Since TATR is an interactive program, it performs only in response to user stimulus. As the program executes, it periodically provides the user with an opportunity either to give an instruction or to enter a report. In the following subsections we describe all instructions and reports that can be entered by the user and provide an annotated trace of an exemplary plan generation showing typical usages and results of several instructions and reports. A complete transcript of user-TATR interaction is exhibited in the Appendix.

Instructions

Instructions allow the user to direct program actions, display desired information, and modify the plan generated by program heuristics and logic. The instructions are of two basic types, display and command. The display instructions generally cause tables describing the current status of the plan or airfields to be printed. The command instructions tell the program how to proceed with the attacks, usually overriding the attack plans devised by the program. For example, the user can tell the program how many strikes to make at one time or what target elements on what airfield to include in the plan.

The instructions available for the user are as follows:

Go compute.

purpose: Used to initiate the plan-generation function.

result: TATR executes the plan-generation function and displays a listing of the ATL.

Let the number of strikes be N.

purpose: Used to restrict the number of airfields on the ATL to include in the plan. The default is the total ATL.

example: Let the number of strikes be 6.

result: The plan is limited to the first six airfields on the ATL.

Go display_targets.

purpose: Used to display more detailed information about the plan than is provided on the ATL.

result: Each airfield on the ATL is listed with its TI, the target element planned for attack, the number of weapon systems needed, and the AR.

Go display_options for Y.

purpose: Used to display the full set of attack options for airfield Y, i.e., the resulting AR from an attack against each target element on airfield Y.

example: Go display_options for A1.[2]

result: The AR for each target element on A1 (Mirow) is listed.

Go attack X at Y.

purpose: Used to specify an attack to be included in the plan, usually different from one already included by the plan-generation process.

---

[2] For convenience and efficient user actions, airfields are identified by two-character designators. A1 is Mirow, A2 is Parchim, etc. On output, both the designator and the name are displayed.

example: Go attack runways at A1.

result: TATR determines the best weapon system package to attack the runways at A1 (Mirow), calculates the effects of the attack, and includes the attack in the plan.

For every member (m) of the ATL, go display options for m.

purpose: Used to display the full set of attack options for every airfield on the ATL.

result: The AR for each target element is listed for every airfield on the ATL.

Go display_opstats.

purpose: Used to display the operational status of the target elements on the airfields.

result: A table showing the operational status of each target element on every airfield.

Let the desired_DE for X be N.

purpose: Used to specify the desired damage expectancy (N) to be used to determine the number of weapon systems needed to attack target element X. The default is .8 for runways and .7 for all other target elements.

example: Let the desired_DE for runways be .9.

result: When the weaponeering function calculates the number of weapon systems (aircraft and munitions) needed to attack runways, it will select the minimum number to achieve a damage expectancy of .9 rather than whatever damage expectancy was previously being used.

Quit.

purpose: Used to terminate program execution.

<u>Reports</u>

The reports the user can enter are all updates to the database. Thus, the user might report that Mirow has characteristics of a key unit

airfield, that the number of Floggers at Brandenburg/Priest is actually 29, that the observed operational status for maintenance at Templin is .6, or that the number of aircraft that actually arrived over a target during a previous attack was 4. The user can also change or delete any data element in the database or add new elements, including entire enemy airfields.

User inputs in response to a program request for reports are optional. If information concerning actual or anticipated changes in the status of database elements is not provided by the user, the program continues, making the best assumptions it can under the circumstances.

Reports pertaining to dynamic updating in response to operations and intelligence reports of battle actions are illustrated below.

Let the number of X at Y be N.

purpose: Used to update the quantity (N) of a target element (X) at an airfield (Y).

example: Let the number of Floggers at A1 be 26.

result: The number of Floggers at Mirow is set at 26.

Let the reconrate of X at Y be <N1,N2,...Ni>.

purpose: Used to update the capability reconstitution rate (the rate at which capability is estimated to be restorable after damage from an attack) of a target element (X) at an airfield (Y). N1 is the reconstitution percentage for the first day after damage, N2 for the second, ... Ni for the ith and final day, i.e., in i days the target element X at Y is expected to be reconstituted to full operational capability.

example: Let the reconrate for munitions at A1 be <0,.5,.5>.

result: The reconstitution percentages for munitions at Mirow are set in the database to 0 (none) on the first day after munitions were damaged by an attack, 50% on the second day and 50% on the third day.

Let the observed_opstat for X at Y be N.

purpose: Used to update the observed operational status (N) of a target element (X) at an airfield (Y).

example: Let the observed_opstat for munitions at A1 be .6.

result: The operational status for the munitions at Mirow is set to .6 (60% of that required to sustain full operational capability of the forces located there).

Let the reconperiod for X at Y be N.

purpose: Used to accompany observed operational status reports to indicate the apparent progress through the reconstitution cycle for a target element (X) at an airfield (Y).

example: Let the reconperiod for munitions at A1 be 2.

result: The reconstitution period for the munitions at Mirow is set at 2, i.e., reconstitution of munitions support capability is considered to be in its second day.

Let the number of aircraft over target be N.

purpose: Used in conjunction with a specified attack to indicate that N aircraft actually attacked the target element. Often, N is different from the planned number of aircraft due to sortie-generation constraints, aborts, and attrition.

example: Let the number of aircraft over target be 9.

result: When projecting the effects of the attack, the program will use nine aircraft instead of the planned number.

Assert Y is a key_unit airfield.

purpose: Used to designate key unit airfields.

example: Assert A1 is a key_unit airfield.

result: Mirow will be considered as a key unit airfield by the plan-generation function.

Let the ceiling at Y be N ft.

purpose: Used to update the ceiling at an airfield (Y).

example: Let the ceiling at A1 be 4500 ft.

result: The ceiling at Mirow will be set at 4500 ft.

Let the visibility at Y be N nm.

purpose: Used to update the visibility at an airfield (Y).

example: Let the visibility at A1 be 8 nm.

result: The visibility at Mirow will be set at 8 n mi.

Print Priority

As the TATR program executes, it produces a running commentary or explanation of the calculations it is making. The extent or completeness of this explanation can be controlled by the "print priority" instruction. With the print priority set to 10, a very detailed explanation is given (everything that could be printed is printed). When the print priority is set to 1, results are given without explanation. As the print priority is increased from 1 to 10, the detail of the explanation becomes greater. The print priority can be changed whenever either reports or instructions are called for by the program.

The form of the print priority specification is

Let the pp be N.

example: Let the pp be 3.

The default print priority is 9, which provides considerable detail.

Example User-Interface Trace

The following annotated listing of a trace of the information displayed on the user's terminal during an abbreviated plan-generation

run illustrates the outputs and the interface actions discussed above. TATR's outputs are shown in normal type, the user's inputs are shown in boldface type, and the authors' comments are shown in brackets.

Go compute.

[This user input starts program execution.]

Date: day 1

Ready for report: **Let the opstat for shelters at A3 be .6.**
Let the reconperiod for shelters at A3 be 0.
Let the opstat for revetments at A6 be .3.
Let the reconperiod for revetments at A6 be 0.
Assert A9 is a key_unit airfield.
Let the pp be 3.

[The user updates the database using currently obtained information. The pp is the print priority--the lower the number, the less trace produced.]

Report understood.

The ATL is (A1 A7 A2 A8 A9*).

[These are the airfields planned for attack, ordered by achieve ratio. The asterisk indicates that A9 is a key unit airfield.]

Ready for instructions: **go display_targets.**

AIRFIELDS

TI	Target Name	BE Number	Element	No. of Aircraft	AR
2088	A1: Mirow	9015	mun	3	696
1800	A7: Drewitz	9029	st-a	3	600
1728	A2: Parchim	9017	st-a	3	576
1908	A8: Falkenberg	9030	st-b	6	318
780	A9: Finsterwalde	9032	st-b	6	130

[The user asks for a more informative display of the airfields on the ATL. The result is a table showing for each airfield its TI, name, BE-number, target element with highest AR, and number of aircraft required to destroy that target. The target elements abbreviated are mun/munitions, st-a/above_ground_pol_storage_sites, st-b/below_ground_pol_storage_sites.]

go display_options for A1.

```
            A1 (9015) :  Mirow
    Achieve Ratio           Element
         67             84  AIRCRAFT
        120              2  MAINTENANCE_HARD
        130              4  MAINTENANCE_SOFT
        522              4  ABOVE_GROUND_POL_STORAGE_SITES
        348              3  BELOW_GROUND_POL_STORAGE_SITES
        696              2  MUNITION_SITES
         15             40  SHELTERS
        120              4  REVETMENTS
```

[The user asks for a more informative display of airfield A1. The result is a table showing the AR for each target element of the airfield and the number of those elements at the airfield.]

Let the number of strikes be 1.

[The user states that the number of airfields attacked should be 1 rather than the default number (the length of the ATL).]

Instructions understood.

The achieve ratio for A1 is 696,
(using MUNITION_SITES as the target).

Now calculating possible weapon packages
for attacking MUNITION_SITES at A1.

POSSIBLE WEAPON PACKAGES

Weapack	Weapon System	Aircraft	DE	Attrition	Del Tactics	AR
WEAPACK #242	F-111X/2	3	.71	0.29	LOW	696
WEAPACK #243	F-4X/4	4	.86	0.63	HIGH	522
WEAPACK #244	F-111X/1	5	.75	0.49	LOW	417
WEAPACK #245	F-4X/1	12	.73	1.91	HIGH	174

[The program picks the airfield on the ATL with the highest AR (A1 with target element munition sites) and shows the possible weapon packages that were considered for attacking the target element in question and the ARs resulting from the use of each weapon package. The line of the table containing the highest AR shows the weapon package recommended by the program (i.e., 3 F-111Xs carrying standard conventional load 2 and using a low-angle delivery tactic).]

Ready for report: **Let the number of aircraft over target be 2.**

[The user states that only 2 of the 3 attacking aircraft actually arrived over the target.]

Report understood.

Reports indicate only 2 aircraft over the MUNITION_SITES, thus the probability of destroying the MUNITION_SITES has been estimated to be .57.

[The probability of destroying the munitions sites would have been .71 (see table) if all three aircraft had arrived over the target.]

Airfield Operational Status

	runw	air	mn-h	mn-s	st-a	st-b	mun	shel	revt	sams
A1	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(.57)	(0)	(0)	(0)
A2	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	.6	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A5	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A6	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	.3	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A7	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A8	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A9	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)

[A table is now displayed showing the updated operational status for each target element of each airfield in the database. The unparenthesized numbers represent the hypothesized operational status of each target element, while the parenthesized numbers represent the probability that the target element has been completely destroyed.]

Ready for report: **Quit**

III. THE TATR PROGRAM

OVERVIEW

The main elements of the TATR program structure are listed in Table 1, with the contents of these elements shown to the right of each element. Each of the elements is described in the paragraphs that follow.

Table 1

TATR PROGRAM ELEMENTS

Main Elements	Contents			
Target database	Airfields	Airfield target elements	Defense missile sites	Weather
Operational status	Current status	Forecast status	Battle damage input	Target reconstitution system
Rules	General rules	Policy file	Users file	
Weaponeering	Probability of arrival tables	Probability of damage tables	Computation functions	
Displays	Air target list	Weapon systems packages	Strike results	Target operational status

Target Database

The target database contains a selected set of enemy airfields and defense missile (SAM) sites extracted from an unclassified exercise database used by the Air Force in the Air/Ground Operations School. In addition to general airfield data, each airfield entry contains detailed information about the important target elements (e.g., runways, aircraft, fuel storage) located at the airfield. Weather forecasts also are included because TATR can limit the weapon systems it considers to those whose delivery parameters are below the ceiling and visibility forecast for a target. An example airfield database entry was shown in Fig. 1. Based on user inputs, the program dynamically updates the target database to reflect rapidly changing conditions to be expected in a combat environment.

Operational Status

The operational-status element reflects the current and forecast status of airfields, the target elements of the airfields, and the SAM sites. The operational status (opstat) is expressed as the proportion (from 0 to 1) of the airfield or SAM site potential operational capability that can be supported. The opstat of an airfield is the minimum opstat of the target elements at that airfield. Battle damage resulting from each strike may be entered by the user, representing an input from an intelligence battle damage report. If such an input is not made, the program assumes all planned strikes occur and achieve the damage estimated. The opstat system also incorporates provisions for the reconstitution of targets based on estimates of the improvement in a

target's capability expected to be achieved in each time period after a strike. This increment of improvement is applied to a target's opstat each time period unless it is overridden by an input of confirmed target status.

Rules

TATR is programmed primarily in the ROSIE programming language. The program consists of English-like production rules organized in logical, procedural rulesets. The organization and form of the rules are designed to facilitate the user's comprehension of the program flow and logic. The main body of rules, referred to as general rules in Table 1, perform the primary tasks of developing the attack plan, interacting with the user, dynamically updating data files, and controlling the sequence of program events. Although general rules can be readily modified, as can any TATR rule or database item, we consider them to be firm in the sense that a user would not normally change them for any particular operational run. To permit needed operational flexibility, we have identified two sets of rules and parameters called the policy file and the user file.

The members of the policy and user file sets are those rules and parameters that would normally be changed by a user to account for situational variation, command guidance and direction, and individual targeteer approaches. The policy file contains items that targeteer users have no independent authority to establish or change and would be procedurally bound not to. Policies and directions from higher authorities (e.g., command, theater, national) fall into this category

and might include rules of engagement, political and geographical limitations, and weapon system employment constraints. The user file contains items that targeteer users have complete control over in interacting with TATR to develop an attack plan. These include attack objectives, desired damage expectancy, defense suppression package formulation, and rules, data, and parameters for ATL generation which allow for exploring variations in ATLs under different conditions.

An example of rules written in the ROSIE language is shown in Fig. 2. The ruleset determines whether a "given_aircraft" (previously identified by other rules in the program) can be used against either a "given_airfield" or a "proposed target" (also previously identified

```
To determine given_aircraft is permitted to be employed against
given_airfield:

[1] If the given_aircraft is equal to A-10X
   and the distance to the given_airfield is greater than 60 nm,
   let the current_justification be the string 1/E2/"the distance to ",
       the given_airfield, " exceeds 60 nm"1/T2/
   and conclude false [i.e., the given_aircraft is not permitted to
                       be employed against the given_airfield].

[2] If the given_aircraft is equal to F-111X
   and the proposed target is not contained in
           <maintenance_hard, munition_sites, sam_sites>,
   let the current_justification be the string 1/E2/"the target at ",
        the given_airfield,
        " is not maintenance_hard, munition_sites or sam_sites"1/T2/
   and conclude false [i.e., the given_aircraft is not permitted to
                       be employed against the given_airfield].

[3] Conclude true [that the given_aircraft is permitted to be
                   employed against the given_airfield].

End.
```

Fig. 2--Example TATR Ruleset

by the program). This ruleset would probably be a member of the policy file. It reflects policies concerning the use of certain aircraft types against certain targets. The policies prohibit the use of the A-10X against airfields more than 60 nautical miles from the battle area and the use of the F-111X to attack other than specified target elements. The exam rules also produce message text explaining why the decision was made not to employ a particular aircraft. The user can have the messages displayed if he is interested.

Weaponeering

The weaponeering element determines which aircraft types, munition loads, and delivery tactic combinations are effective against a given target element and calculates how many aircraft are required to achieve a desired damage expectancy against that target element. Effective combinations are determined by rules such as those shown in Fig. 2. The calculation routine considers the probability of the aircraft arriving at the target and the probability of the aircraft that arrive damaging the target with the munitions being carried and the delivery tactic used. These probabilities are provided to the program in tabular form. The computational procedure used is greatly simplified, compared with damage computation routines normally used by the Air Force. However it provides sufficient weaponeering capability for our immediate needs. In a real operational implementation of TATR, we would interface TATR with an official, existing Air Force weaponeering program.

The weaponeering subroutine that calculates weapons effects is programmed in INTERLISP. Determination of which aircraft types,

munitions loads, and delivery tactics to apply is made by ROSIE rules in the main body of the TATR program. All data reflecting aircraft capability are constructive and unclassified.

Displays

Since TATR is an interactive program, all outputs are provided to the user online at his or her own terminal. Provision is made for online displays to be saved and printed in hardcopy form if the user is interacting via a video display terminal. The information displayed includes the items listed in Table 2 and any target database entry. Displays are demonstrated in the listings of example user interactions in Section II and the Appendix.

TATR LOGIC FLOW

The TATR program generates an airfield attack plan following six major steps:

- Develop an initial airfield target list (ATL) of highest-threat airfields.
- Weaponeer target elements at the ATL airfields.
- Form strike packages of airfield attack aircraft.
- Determine the achieve ratio (benefit-to-cost ratio) for each airfield/strike package combination.
- Order and display a suggested ATL.
- Interact with the user to develop a final ATL.

Each step is discussed below.

Determine an Initial ATL

Calculate threat indices. A threat index (TI) is calculated for each airfield in the target database. The calculation sums the threat contribution from each type of aircraft located at the airfield. These contributions are computed as the product of four factors: (1) the quantity of that type of aircraft located on the airfield, (2) the relative value index, called the relval, of that type of aircraft[1], (3) the distance-to-target factor for that type of aircraft that adjusts for the range of the aircraft and the location of the airfield, and (4) the airfield opstat.

Select and order airfields. First, airfields having a TI within 20 percent of the largest TI of all the airfields are considered to be the highest-threat airfields and are included in the ATL. Second, airfields having a TI within 5 percent of the smallest TI of the airfields first selected are added to the ATL.[2] Third, the airfields are ordered by descending TI. Fourth, airfields on the ATL that have been designated key unit (KU) airfields[3] are advanced, in TI order, to the top of the ATL.

---

[1] Aircraft relvals reflect the user's perception of aircraft sortie rates and all aspects of operational capability, such as delivery accuracy, munitions flexibility, and night and adverse weather capability. The values are relative among aircraft performing the same mission against friendly forces, e.g., offensive counterair, defensive counterair, or offensive air support.

[2] The 20 percent and 5 percent thresholds are the program default thresholds. The user can reset them to any desired values at the initiation of or during program execution.

[3] Unique characteristics of some airfields may cause a TATR user to believe that the TATR formula for the TI alone does not adequately reflect the threat from those airfields. Intelligence may have determined that the combat units on an airfield are historically superior to similar units on other airfields, or that the airfield is a maintenance

Last, airfields not already on the ATL that have been designated KU airfields are added to the bottom of the ATL in descending order of their TIs.

The resulting set of airfields is the initial ATL, selected and ordered by targeteer rules for threat determination.

Weaponeer Target Elements

The second step in the plan-generation process is to determine the best aircraft, munitions load, and delivery tactic combination for attacking the relevant target elements on each of the ATL airfields. The relevancy of a target element is determined by the objective of the attack.

The user has several strike objectives from which to choose--interrupt operations, aircraft attrition, sortie attrition, or any combination of the three. The selection determines which target elements would be attacked. To interrupt operations, only runways are attacked. For the aircraft attrition objective, uncovered aircraft, revetments, and shelters are attacked. For the sortie attrition objective, uncovered aircraft and all support functions are attacked. These options represent the range of choices desired by targeteers in length of impact and degree of damage. The program default objective is sortie attrition.

Determine feasible options. For each relevant target element, the program identifies feasible combinations of aircraft, munitions load, and

---

depot, or that it has essential logistics elements, nuclear weapons, or other strategically important assets. Therefore, TATR provides a means to specify these airfields as KU airfields.

delivery tactic by using data that represent weapons effects calculations from operational tests. The feasible combinations are then screened by rules (such as those shown in Fig. 2) that reflect policy, user, or operational (e.g., range) constraints.

Apply weaponeering subroutine. Feasible combinations that survive the screening are submitted to a weaponeering subroutine. Figure 3 shows the inputs and outputs of the weaponeering subroutine. The user may set the desired damage expectancy (default is .7). The actual damage expectancy usually will exceed the desired damage expectancy because the weaponeering subroutine always satisfies the desired DE and applies only integer numbers of aircraft.

Select best combination. At present, the best combination is considered to be the one that requires the fewest aircraft to achieve the desired damage expectancy. This selection criterion will undergo modification in the next program implementation to reflect perceived relative worth of aircraft, alternative uses, and scarcity.

Form Strike Packages

The program next determines how many of which types of aircraft are needed to attack each airfield on the ATL. The present version of TATR uses only the aircraft type that can achieve the desired damage expectancy with the fewest aircraft. That aircraft type and number, called the attack force, forms the strike package.

Logic has been developed to determine defense suppression and air defense escort aircraft and to batch attack forces for efficient escort utilization. This logic will be included in the next program

SOURCE OF INFORMATION		INPUT		OUTPUT
ATL	------>	AIRFIELD		AIRFIELD
TATR RULES	------>	TARGET ELEMENT		TARGET ELEMENT
TATR RULES	------>	AIRCRAFT/ MUNITIONS/ TACTIC	WEAPONEERING SUBROUTINE	AIRCRAFT/ MUNITIONS/ TACTIC
USER	----->	STRIKE OBJECTIVE		QUANTITY OF AIRCRAFT
USER	----->	DESIRED DAMAGE EXPECTANCY		EXPECTED ATTRITION
OPSTAT	----->	PREVIOUS STRIKE RESULTS		DESIRED DAMAGE EXPECTANCY
				ACTUAL DAMAGE EXPECTANCY

Fig. 3--Weaponeering Subroutine

implementation. Then the strike package will consist of the attack force(s) and the support aircraft.

Calculate Achieve Ratios

The fourth step in the plan-generation process calculates an achieve ratio (AR) for each target element on each airfield. The ARs reflect relative benefit-to-cost tradeoffs achieved by attacking a

particular airfield with a particular strike package. The present version of TATR calculates the AR by dividing the number of aircraft in the strike package into the reduction in the airfield TI resulting from the attack.

Present TATR logic determines the airfield TI reduction from battle damage to be either 0 or the total TI. If the projected damage expectancy of the target element remains below a user-specified threshold (the default is .7) following a planned attack, the heuristic assumption is that the target has survived; hence, the TI reduction is 0. If the projected damage expectancy exceeds the threshold, the heuristic assumption is that the target was destroyed; hence, the TI reduction is total airfield TI.

Order and Display ATL

The fifth step in the plan-generation process adjusts the ATL order to reflect the ARs and displays it to the user. The final order is primarily descending by AR for an airfield; however, airfields whose ARs compare closely in value retain their original ordering. The AR used for an airfield is the largest AR of the target-element/strike-package combinations associated with the airfield.

User Interaction

The final step in the plan-generation process permits direct interaction with and involvement by the user. Using the interactive instructions and reports described in Section II and illustrated in the Appendix, the user can directly modify the plan or investigate the

effect of changes to operational conditions and/or parameters assigned to the user file. User interaction will result in TATR reaccomplishing one or more of the previous five steps in the plan-generation process.

IV. CONCLUDING REMARKS

We consider TATR to be in the very initial stage of implementation. The approach to airfield prioritization and target-element selection establishes tradeoffs among key factors--enemy threat, friendly capability, expected outcomes, and cost. This approach and the program structure and heuristics by which it is implemented represent an initial attempt of actual Air Force targeteers to describe the targeting procedures they apply. Other targeteers who have reviewed TATR have generally concurred with the approach and the key factors. However, as usually occurs in the implementation of an expert system, other, seemingly better heuristics and logic structures have now surfaced.

Particular interest is centered on the dominant quantitative aspect of the present TATR decision process. We have three concerns. One is that the present quantitative relationships are so simplistic that they probably provide poor representations of real situations. The second is that, at present, quantitative focus may overstate our capability to determine valid numeric inputs and quantitative relationships reflecting enemy operations and support activities, either before or during a conflict. The final concern is that although the current tactical air targeting trend aims toward determining analytically based solutions, current practice remains dependent on qualitative and heuristic judgments.

To overcome these and other weaknesses in TATR, we are using the present version of TATR as a vehicle for interfacing with the targeting community to evolve a knowledge base that has increasing insight into

non-quantitative targeting techniques used by today's tactical targeteers. Our efforts now focus on developing a program structure and rules that are more reflective of these techniques and implementing them in the next version of TATR.

Appendix

TRANSCRIPT OF USER-TATR INTE· ·TION

The following is an annotated video display listing resulting from a user running the TATR program. TATR's outputs are shown in normal type, the user's inputs are shown in boldface, and the authors' annotations are shown in brackets.

Go compute.

[This user input starts program execution.]

Date: day 1

Ready for report: **Let the opstat for shelters at A3 be .6.**
Let the reconperiod for shelters at A3 be 0.
Let the opstat for revetments at A6 be .3.
Let the reconperiod for revetments at A6 be 0.
Assert A9 is a key_unit airfield.
Let the pp be 3.

[The user updates the database using currently obtained information. The pp is the print priority--the lower the number, the less trace produced.]

Report understood.

The ATL is (A1 A7 A2 A8 A9*).

[These are the airfields planned for attack, ordered by achieve ratio. The asterisk indicates that A9 is a key unit airfield.]

Ready for instructions: **go display_targets.**

AIRFIELDS

TI	Target Name	BE Number	Element	No. of Aircraft	AR
2088	A1: Mirow	9015	mun	3	696
1800	A7: Drewitz	9029	st-a	3	600
1728	A2: Parchim	9017	st-a	3	576
1908	A8: Falkenberg	9030	st-b	6	316
780	A9: Finsterwalde	9032	st-b	6	130

[The user asks for a more informative display of the airfields on the ATL. The result is a table showing for each airfield its TI, name, BE-number, target element with highest AR, and number of aircraft required to destroy that target. The target elements abbreviated are mun/munitions, st-a/above_ground_pol_storage_sites, st-b/below_ground_pol_storage_sites.]

go display_options for A1.

A1 (9015) : Mirow

Achieve Ratio		Element
67	84	AIRCRAFT
120	2	MAINTENANCE_HARD
130	4	MAINTENANCE_SOFT
522	4	ABOVE_GROUND_POL_STORAGE_SITES
348	3	BELOW_GROUND_POL_STORAGE_SITES
696	2	MUNITION_SITES
15	40	SHELTERS
120	4	REVETMENTS

[The user asks for a more informative display of airfield A1. The result is a table showing the AR for each target element of the airfield and the number of those elements at the airfield.]

Let the number of strikes be 1.

[The user states that the number of airfields attacked should be 1 rather than the default number (the length of the ATL).]

Instructions understood.

The achieve ratio for A1 is 696,
(using MUNITION_SITES as the target).

Now calculating possible weapon packages
for attacking MUNITION_SITES at A1.

POSSIBLE WEAPON PACKAGES

Weapack	Weapon System	Aircraft	DE	Attrition	Del Tactics	AR
WEAPACK #242	F-111X/2	3	.71	0.29	LOW	696
WEAPACK #243	F-4X/4	4	.86	0.63	HIGH	522
WEAPACK #244	F-111X/1	5	.75	0.49	LOW	417
WEAPACK #245	F-4X/1	12	.73	1.91	HIGH	174

[The program picks the airfield on the ATL with the highest AR (A1 with target element munition sites) and shows the possible weapon packages that were considered for attacking the target element in question and the ARs resulting from the use of each weapon package. The line of the table containing the highest AR shows the weapon package recommended by the program (i.e., 3 F-111Xs carrying standard conventional load 2 and using a low-angle delivery tactic).]

Ready for report: **Let the number of aircraft over target be 2.**

[The user states that only 2 of the 3 attacking aircraft actually arrived over the target.]

Report understood.

Reports indicate only 2 aircraft over the MUNITION_SITES,
thus the probability of destroying the MUNITION_SITES
has been estimated to be .57.

[The probability of destroying the munitions sites would have been .71 (see table) if all three aircraft had arrived over the target.]

Airfield Operational Status

	runw	air	mn-h	mn-s	st-a	st-b	mun	shel	revt	sams
A1	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(C)	(0)	(0)	(0)	(.57)	(0)	(0)	(0)
A2	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	.6	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A5	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A6	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	.3	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A7	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(.0)	(0)	(0)	(0)	(0)	(0)
A8	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A9	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)

[A table is now displayed showing the updated operational status for each target element of each airfield in the database. The unparenthesized numbers represent the hypothesized operational status of each target element, while the parenthesized numbers represent the probability that the target element has been completely destroyed.]

Date: day 2.

Ready for report: **Let the top_ATL_percentage be .3.**
Let the pp be 5.

[The user indicates that when calculating the ATL the program should include all airfields whose TIs are within 30% of the highest TI (rather than the default 20%). The user also redefines the print priority so that more of the program's calculations and explanations will be displayed. Note that changing the pp does not affect the number of calcuations made by the program, only the display of those calculations.]

Report understood.

A3's reconperiod for SHELTERS is now 1.
Adding .2 to .6 (the opstat for SHELTERS at A3).
The opstat for SHELTERS at A3 is now .8.

A6's reconperiod for REVETMENTS is now 1.
Adding .1 to .3 (the opstat for REVETMENTS at A6).
The opstat for REVETMENTS at A6 is now .4.

A7's reconperiod for ABOVE_GROUND_POL_STORAGE_SITES is now 1.
Adding 0 to 0.0 (the opstat for ABOVE_GROUND_POL_STORAGE_SITES at A7).
The opstat for ABOVE_GROUND_POL_STORAGE_SITES at A7 is now 0.0.

[Here we see the damaged target elements recovering based on a step function defined in the database as the reconrate for each element.]

The current_TI for A1 is 2088
The current_TI for A2 is 1728
The current_TI for A3 is 936
The current_TI for A4 is 1305
The current_TI for A5 is 1488
The current_TI for A6 is 806
The current_TI for A7 is 0
The current_TI for A8 is 1908
The current_TI for A9 is 780

[The TIs for every airfield in the database are shown.]

The ATL is (A2 A1 A8 A5 A9*).

[The ATL is calculated and displayed. Note that not all airfields are included in the ATL, only those whose TIs are within 30% of the highest TI (2088).]

Ready for instructions: **Go display_targets.**

[The user asks for a more informative display of airfields on the ATL.]

AIRFIELDS

TI	Target Name	BE Number	Element	No. of Aircraft	AR
1728	A2: Parchim	9017	st-a	3	576
2088	A1: Mirow	9015	st-a	4	522
1908	A8: Falkenberg	9030	st-b	6	318
1488	A5: Stargarg	9027	mn-s	6	248
780	A9: Finsterwalde	9032	st-b	6	130

Go display_options for A2.

[The user asks for a more informative display of airfield A2.]

A2 (9017) : Parchim

Achieve Ratio		Element
55	117	AIRCRAFT
120	2	MAINTENANCE_HARD
108	4	MAINTENANCE_SOFT
576	3	ABOVE_GROUND_POL_STORAGE_SITES
216	4	BELOW_GROUND_POL_STORAGE_SITES
144	6	MUNITION_SITES
29	20	SHELTERS
120	42	REVETMENTS

Let the number of strikes be 1.
Go display_options for A8.

[The user indicates that only one airfield should be attacked from the ATL (in addition to any explicitly ordered to be attacked) and asks for a more informative display of airfield A8.]

A8 (9030) : Falkenberg

Achieve Ratio		Element
65	78	AIRCRAFT
120	3	MAINTENANCE_HARD
68	6	MAINTENANCE_SOFT
238	6	ABOVE_GROUND_POL_STORAGE_SITES
318	3	BELOW_GROUND_POL_STORAGE_SITES
190	5	MUNITION_SITES
19	30	SHELTERS
120	10	REVETMENTS

Go attack munition_sites at A8.

[The user indicates that an attack should be made on munitions sites at A8. This is in addition to the one attack on an airfield on the ATL.]

Instructions understood.

Now calculating possible weapon packages
for attacking MUNITION_SITES at A8.

The MAV is not permitted for use against MUNITION_SITES.

The weapon systems that can be used against MUNITION_SITES are:
F-111X/2 F-4X/4 F-111X/1 F-4X/1 A-10X/2

A-10X's are not permitted to fly against airfield A8
(since the distance to A8 exceeds 60 nm).

[Here the program indicates why some weapon systems were ruled out during the weapon-package calculation.]

POSSIBLE WEAPON PACKAGES

Weapack	Weapon System	Aircraft	DE	Attrition	Del Tactics	AR
WEAPACK #263	F-111X/2	10	.74	0.99	LOW	190
WEAPACK #264	F-4X/4	11	.73	1.75	HIGH	173
WEAPACK #265	F-111X/1	17	.72	1.69	LOW	112
WEAPACK #266	F-4X/1	42	.70	6.71	HIGH	45

[The program picks the munition sites at A8 as the target (as instructed by the user) and shows the possible weapon packages that were considered for attacking the target element in question and the ARs resulting from the use of each weapon package. Again, the line of the table containing the highest AR shows the weapon package recommended by the program (i.e., 10 F-111X/2s using a low-angle delivery tactic).]

Ready for report: **Let the number of aircraft over target be 8.**

[The user indicates that only 8 of the 10 aircraft actually arrived over the target.]

Report understood.

Reports indicate only 8 aircraft over the MUNITION_SITES,
thus the probability of destroying the MUNITION_SITES
has been estimated to be .48.

The achieve ratio for A2 is 576,
(using ABOVE_GROUND_POL_STORAGE_SITES as the target).

Now calculating possible weapon packages
for attacking ABOVE_GROUND_POL_STORAGE_SITES at A2.

The weapon systems that can be used against ABOVE_GROUND_POL_STORAGE_SITES are:
F-4X/3 A-10X/2 A-10X/1 F-111X/2 F-4X/4 F-111X/1 F-4X/1

F-111X's are not permitted to fly against airfield A2
(since the target at A2 is not maintenance_hard, munition_sites or sam_sites).

[Again, the program indicates why some weapon systems were ruled out during the weapon-package calculation.]

POSSIBLE WEAPON PACKAGES

Weapack	Weapon System	Aircraft	DE	Attrition	Del Tactics	AR
WEAPACK #275	F-4X/3	5	.73	0.60	LOW	345
WEAPACK #276	A-10X/2	5	.76	0.39	LOW	345
WEAPACK #277	A-10X/1	3	.75	0.23	LOW	576
WEAPACK #278	F-4X/4	4	.75	0.63	HIGH	432
WEAPACK #279	F-4X/1	6	.73	0.95	HIGH	288

[The program picks the above-ground pol storage sites at A2 to attack, since this target has the highest AR on the ATL. The program shows the possible weapon packages that were considered for attacking the target element in question and the ARs resulting from the use of each weapon package. Again, the line of the table containing the highest AR shows the weapon package recommended by the program (i.e., 3 A-10X/1s using a low-angle delivery tactic).]

Ready for report:
No report made.

The probability of destroying the ABOVE_GROUND_POL_STORAGE_SITES at A2 has been estimated to be .75.
The new opstat for the ABOVE_GROUND_POL_STORAGE_SITES has been assumed to be 0, (since the probability of target destruction is .7 or greater).

Airfield Operational Status

	runw	air	mn-h	mn-s	st-a	st-b	mun	shel	revt	sams
A1	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(.57)	(0)	(0)	(0)
A2	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(.75)	(0)	(0)	(0)	(0)	(0)
A3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	.8	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A5	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A6	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	.4	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A7	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(.75)	(0)	(0)	(0)	(0)	(0)
A8	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(.48)	(0)	(0)	(0)
A9	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)

[Again, the table showing the updated operational status for each target element of each airfield in the database is displayed.]

Date: day 3.

Ready for report:
No report made.

A2's reconperiod for ABOVE_GROUND_POL_STORAGE_SITES is now 1.
Adding 0 to 0.0 (the opstat for ABOVE_GROUND_POL_STORAGE_SITES at A2).
The opstat for ABOVE_GROUND_POL_STORAGE_SITES at A2 is now 0.0.

A3's reconperiod for SHELTERS is now 2.
Adding .2 to .8 (the opstat for SHELTERS at A3).
The opstat for SHELTERS at A3 is now 1.0.

A6's reconperiod for REVETMENTS is now 2.
Adding .1 to .4 (the opstat for REVETMENTS at A6).
The opstat for REVETMENTS at A6 is now .5.

A7's reconperiod for ABOVE_GROUND_POL_STORAGE_SITES is now 2.
Adding .1 to 0.0 (the opstat for ABOVE_GROUND_POL_STORAGE_SITES at A7).
The opstat for ABOVE_GROUND_POL_STORAGE_SITES at A7 is now .1.

[Again, we see the damaged target elements recovering based on a step function defined in the database as the reconrate for each element.]

The current_TI for A1 is 2088
The current_TI for A2 is 0
The current_TI for A3 is 1170
The current_TI for A4 is 1305
The current_TI for A5 is 1488
The current_TI for A6 is 1008
The current_TI for A7 is 180
The current_TI for A8 is 1908
The current_TI for A9 is 780

[Again, the TIs for every airfield in the database are shown.]

The ATL is (A1 A8 A5 A9*).

[The ATL is displayed.]

Ready for instructions: **Go display_targets.**

AIRFIELDS

TI	Target Name	BE Number	Element	No. of Aircraft	AR
2088	A1: Mirow	9015	st-a	4	522
1908	A8: Falkenberg	9030	st-b	6	318
1488	A5: Stargarg	9027	mn-s	6	248
780	A9: Finsterwalde	9032	st-b	6	130

For each member (m) of the ATL, go display_options for m.

A1 (9015) : Mirow

Achieve	Ratio	Element
67	84	AIRCRAFT
120	2	MAINTENANCE_HARD
130	4	MAINTENANCE_SOFT
522	4	ABOVE_GROUND_POL_STORAGE_SITES
348	3	BELOW_GROUND_POL_STORAGE_SITES
448	2	MUNITION_SITES
15	40	SHELTERS
120	4	REVETMENTS

A8 (9030) : Falkenberg

Achieve	Ratio	Element
65	78	AIRCRAFT
120	3	MAINTENANCE_HARD
68	6	MAINTENANCE_SOFT
238	6	ABOVE_GROUND_POL_STORAGE_SITES
318	3	BELOW_GROUND_POL_STORAGE_SITES
110	5	MUNITION_SITES
19	30	SHELTERS
120	10	REVETMENTS

A5 (9027) : Stargarg

Achieve	Ratio	Element
42	96	AIRCRAFT
120	1	MAINTENANCE_HARD
248	2	MAINTENANCE_SOFT
186	6	ABOVE_GROUND_POL_STORAGE_SITES
186	4	BELOW_GROUND_POL_STORAGE_SITES
18	26	MUNITION_SITES
46	12	SHELTERS
120	44	REVETMENTS

A9 (9032) : Finsterwalde

Achieve	Ratio	Element
35	60	AIRCRAFT
120	2	MAINTENANCE_HARD
70	3	MAINTENANCE_SOFT
97	6	ABOVE_GROUND_POL_STORAGE_SITES
130	3	BELOW_GROUND_POL_STORAGE_SITES
20	14	MUNITION_SITES
24	12	SHELTERS
120	5	REVETMENTS

[Here the user instructs the program to display detailed information regarding every member of the ATL.]

Let the number of strikes be 3.

[The user instructs the program to strike 3 airfields from the ATL. Thus the top three will be chosen, i.e., A1, A8, and A5.]

Instructions understood.

[Here is the attack on A1.]

The achieve ratio for A1 is 522,
(using ABOVE_GROUND_POL_STORAGE_SITES as the target).

Now calculating possible weapon packages
for attacking ABOVE_GROUND_POL_STORAGE_SITES at A1.

[For brevity, we have deleted the intervening weapon system output similar to that above.]

POSSIBLE WEAPON PACKAGES

Weapack	Weapon System	Aircraft	DE	Attrition	Del	Tactics	AR
WEAPACK #288	F-4X/3	7	.70	0.84		LOW	298
WEAPACK #289	F-4X/4	6	.73	0.95		HIGH	348
WEAPACK #290	F-4X/1	9	.71	1.43		HIGH	232

Ready for report:
No report made.

The probability of destroying the ABOVE_GROUND_POL_STORAGE_SITES
at A1 has been estimated to be .73.
The new opstat for the ABOVE_GROUND_POL_STORAGE_SITES has been assumed to be 0,
(since the probability of target destruction is .7 or greater).

[Here is the attack on A8.]

The achieve ratio for A8 is 318,
(using BELOW_GROUND_POL_STORAGE_SITES as the target).

Now calculating possible weapon packages
for attacking BELOW_GROUND_POL_STORAGE_SITES at A8.

The MAV is not permitted for use against BELOW_GROUND_POL_STORAGE_SITES.

[For brevity, we have deleted the intervening weapon system output similar to that above.]

POSSIBLE WEAPON PACKAGES

Weapack	Weapon System	Aircraft	DE	Attrition	Del Tactics	AR
WEAPACK #296	F-4X/4	6	.80	0.95	HIGH	318
WEAPACK #297	F-4X/1	21	.71	3.35	HIGH	90

Ready for report: **Let the observed_opstat for below_ground_pol_storage_sites at A8 be .5.**

Report understood.

The new opstat for the BELOW_GROUND_POL_STORAGE_SITES at A8 has been observed to be .5.

[Here is the attack on A5.]

The achieve ratio for A5 is 248, (using MAINTENANCE_SOFT as the target).

Now calculating possible weapon packages for attacking MAINTENANCE_SOFT at A5.

[For brevity, we have deleted the intervening weapon system output similar to that above.]

POSSIBLE WEAPON PACKAGES

Weapack	Weapon System	Aircraft	DE	Attrition	Del Tactics	AR
WEAPACK #300	F-4X/4	12	.73	1.91	HIGH	124
WEAPACK #301	F-4X/1	18	.70	2.87	HIGH	82

Ready for report: **Let the number of aircraft over target be 9.**

Report understood.

Reports indicate only 9 aircraft over the MAINTENANCE_SOFT, thus the probability of destroying the MAINTENANCE_SOFT has been estimated to be .58.

Airfield Operational Status

	runw	air	mn-h	mn-s	st-a	st-b	mun	shel	revt	sam
A1	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(.73)	(0)	(.57)	(0)	(0)	(0)
A2	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(.75)	(0)	(.0)	(0)	(0)	(0)
A3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A5	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(.58)	(0)	(0)	(0)	(0)	(0)	(0)
A6	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	.5	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
A7	1.0	1.0	1.0	1.0	.1	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(.75)	(0)	(0)	(0)	(0)	(0)
A8	1.0	1.0	1.0	1.0	1.0	.5	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(.48)	(0)	(0)	(0)
A9	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)

Date: day 4.

Ready for report: **Let the pp be 9.**

[The user sets the print priority to display every calculation the program makes, together with its explanation to illustrate level of detail available.]

Report understood.

Now updating the clock.

A1's reconperiod for ABOVE_GROUND_POL_STORAGE_SITES is now 1 (up from 0).
Adding 0 to 0.0 (the opstat for ABOVE_GROUND_POL_STORAGE_SITES at A1).
The opstat for ABOVE_GROUND_POL_STORAGE_SITES at A1 is now 0.0.

A2's reconperiod for ABOVE_GROUND_POL_STORAGE_SITES is now 2 (up from 1).
Adding .1 to 0.0 (the opstat for ABOVE_GROUND_POL_STORAGE_SITES at A2).
The opstat for ABOVE_GROUND_POL_STORAGE_SITES at A2 is now .1.

A6's reconperiod for REVETMENTS is now 3 (up from 2).
Adding .2 to .5 (the opstat for REVETMENTS at A6).
The opstat for REVETMENTS at A6 is now .7.

A7's reconperiod for ABOVE_GROUND_POL_STORAGE_SITES is now 3 (up from 2).
Adding .3 to .1 (the opstat for ABOVE_GROUND_POL_STORAGE_SITES at A7).
The opstat for ABOVE_GROUND_POL_STORAGE_SITES at A7 is now .4.

A8's reconperiod for BELOW_GROUND_POL_STORAGE_SITES is now 1 (up from 0).
Adding 0 to .5 (the opstat for BELOW_GROUND_POL_STORAGE_SITES at A8).
The opstat for BELOW_GROUND_POL_STORAGE_SITES at A8 is now .5.

Now calculating the standard TI.

The standard_TI for A1 is 2088
The standard_TI for A2 is 1728
The standard_TI for A3 is 1170
The standard_TI for A4 is 1305
The standard_TI for A5 is 1488
The standard_TI for A6 is 2016
The standard_TI for A7 is 1800
The standard_TI for A8 is 1908
The standard_TI for A9 is 780

[The standard TI is based on the number and value of aircraft at the enemy airfield.]

Now calculating the current TI.

The current_TI for A1 is 0
(2088.0 * 0.0 (the opstat for ABOVE_GROUND_POL_STORAGE_SITES))
The current_TI for A2 is 172
(1728.0 * .1 (the opstat for ABOVE_GROUND_POL_STORAGE_SITES))
The current_TI for A3 is 1170 (1170 * 1.0 (the opstat for RUNWAYS))
The current_TI for A4 is 1305 (1305.0 * 1.0 (the opstat for RUNWAYS))
The current_TI for A5 is 1488 (1488.0 * 1.0 (the opstat for RUNWAYS))
The current_TI for A6 is 1411 (2016.0 * .7 (the opstat for REVETMENTS))
The current_TI for A7 is 720
(1800.0 * .4 (the opstat for ABOVE_GROUND_POL_STORAGE_SITES))
The current_TI for A8 is 954
(1908.0 * .5 (the opstat for BELOW_GROUND_POL_STORAGE_SITES))
The current_TI for A9 is 780 (780.0 * 1.0 (the opstat for RUNWAYS))

[The current TI is based on the standard TI and the current operational status of each airfield.]

Now ordering the airfields by TI.

Now refining the ordered airfield_list.

Now reordering the ATL using key units.

Now reordering the ATL by AR.

The ATL is (A4 A3 A6 A5 A9*).

Ready for instructions: **Quit.**

REFERENCES

1. Callero, Monti, Daniel Gorlin, Frederick Hay· ·Roth, and Lewis Jamison, Toward an Expert Aid For Tactical Air Targeting, The Rand Corporation, N-1[illegible]-ARPA, January 1981.

2. Fain, J., D. Gorlin, F. Hayes-Roth, S. Rosenschein, H. Sowizral, and D. Waterman, The ROSIE Language Reference Manual, The Rand Corporation, N-1647-ARPA, December 1981.

3. Fain, J., F. Hayes-Rot·, H. Sowizral, and D. A. Waterman, Programming in ROSIE: An Introduction by Means of Examples, The Rand Corporation, N-1646-ARPA (forthcoming).

PRECEDING PAGE BLANK-NOT FILMED